Applying Problem-Solving Heuristics to a Freshman Engineering Course

Shari J. Kimmel[1], Fadi P. Deek[2], Howard S. Kimmel[2]
[1]Penn State Berks-Lehigh Valley College and Lehigh University
[2]New Jersey Institute of Technology

Abstract

Many students enter undergraduate engineering programs lacking basic problem solving skills. We have adapted the problem solving heuristics originally used in a computer science environment to an introductory engineering class to help freshman engineering students develop these skills. The introductory Engineering Design and Graphics course at Penn State - Berks Campus exposes students to conventional drafting techniques, computer graphics and engineering design.

The basic heuristic involves identifying the precise problem from a vague problem statement and then subdividing it into smaller tractable parts. Identifying the problem statement, analyzing possible solutions and presenting the final result are important aspects of the engineering design component. The solid modeling component of the course lends itself to giving students practice dividing a large problem into smaller segments for which a solution method can be more easily determined and tested. The problem solving exercises taught students to analyze the problem and think about the solution instead of rushing to implement an ill-considered solution. This assignment improved their performance compared to another section of the same course which did not include specialized problem solving activities. However, to properly interpret an unstructured problem statement, students would benefit from starting with a guided instructional approach.

The students' perceptions about the value of this method vary. Some students think that many of these problems are too easy to make the extra work involved in a formal problem solving method worthwhile, but most students see the value of this formal method for solving complicated problems. A primary obstacle to implementing this method in the classroom is that students prefer to avoid the initial additional effort required by a formal problem solving strategy.

I. Introduction

Implementation of freshman engineering design courses has become commonplace, so as to provide the first year students with early experiences in engineering. Universities have used many different approaches in the development and implementation of these courses, some of which have been summarized by Richardson, *et al*.[1]. Regardless of the approach or nature of the course, students enrolled in engineering design classes are expected to be problem solvers. While the characteristics and objectives of such courses may vary, almost all of them include objectives that emphasize the ability to set up and solve problems, and the ability to communicate effectively, both verbally and in writing.[2]

Many students enter undergraduate engineering programs lacking basic problem solving skills. Typically, their method of solving problems involves rote memorization of the material in order to get through the exams, which often consist of simple, well-defined problems, then forgetting it almost immediately thereafter. However, many students would not be able to solve problems that involve changes in wording or context. Though engineering design textbooks often discuss the problem solving aspect of engineering design, they usually provide either an abstract list of steps that are difficult to directly implement or too simplistic a solution process which applies only to already well defined problems.

In the former case, these steps generally include ’defining the problem’ without giving students any guidance about how to make sure the problem is fully defined without being too general or too specific. For instance, in the textbook used during the Fall 2000 semester,[3] steps in the problem solving process are presented in the design section at the beginning of the book and several examples are illustrated. The book’s instructions for ’Problem Identification’ are "First write a statement of the problem and a statement of need. List limitations and desirable features, and make descriptive sketches to better identify the requirements." This type of explanation assumes that students can disassemble and interpret the given information. In our experience, not enough guidance is given for students to apply this method

In the latter case, the problem solving method typically involves steps such as ’defining the problem’, ’listing the givens’, ’writing down the equation’, and ’finding a solution’.[4] This type of textbook then presents an overview of basic principles and equations for several engineering disciplines, including circuit analysis, heat transfer and dynamics. The problems given are very well defined and can generally be solved using a single equation given in that chapter. Though that method gives students practice using a variety of engineering tools and works well for simple textbook problems, they do not learn how to analyze real design problems, which are generally open-ended and unstructured.

Some more useful textbooks exist which guide the student in detail through steps in the design process.[5] One promising example[6] incorporates a fairly detailed analysis of a few design problems throughout the entire book, relating the issues discussed in each chapter to each sample problem and to a problem solving method introduced early in the text. However, no guidance is provided for formulating the problem, i.e. understanding and processing the information in the problem statement. Creating exercises to give students experience with specific aspects of a problem solving method would be very time consuming for the instructor.

Not surprisingly, the typical course in freshman engineering design provides similar superficial treatment for the students. Certainly, one reason is that problem solving skills are perceived to be somewhat vague and difficult to objectively assess and that most faculty probably do not know how or feel comfortable teaching them. "Problem solving" is a phrase that might be found in a course description, but unless it is included in the course objectives and specifically identified as a skill to be mastered in the course, it is, at best, given perfunctory attention in the classroom.[7] In addition, a certain amount of time is needed for students to learn and master these skills. It has been demonstrated that true mastery of skills comes only from practice in the application of the skills in real world situations.[7] The McMaster Problem Solving (MPS) Program involves 120 hours for skills development spread over four required courses.[8] Unfortunately, most universities can not commit that amount of time to the development of these skills.

Problem solving heuristics originally used in an introductory computer science course were adapted to teach problem solving skills to beginning engineering students. The introductory Engineering Design and Graphics course (ED&G 100) at Penn State - Berks Campus exposes students to conventional drafting techniques, computer graphics and engineering design. The typical class consists of mostly first-year and some second year students with a wide range of skills and experience.

During the fall 2000 semester, a section of ED&G 100 with 17 students taught by the first author included writing and problem solving exercises integrated into the curriculum. The major assignment in this course is a group design project in which students apply skills previously learned in the class. In the fall 2000 semester, students were presented with a cart used in an industrial setting. The assignment involved the redesign of the cart to be safer and easier for the workers to use and more space efficient than the original at a reasonable cost.[9]

II. Problem Solving Methodology and Application to Engineering Design

Computer science and engineering share much common ground; problem solving is a fundamental skill necessary to succeed in both disciplines. Many methods of problem solving have been developed. One popular approach was proposed by the mathematician George Polya,[10] who defined a four-step process for solving a problem: understanding the problem, devising a plan, carrying out the plan, and looking back. Each step focuses on a unique aspect of the problem. The first step is concerned with understanding the problem's question and requirements. Comprehension of the problem requires the identification of the goal, the givens, the unknowns, the conditions, the constraints, and their relationship. Devising the plan is the outline and refinement of a potential solution to the problem. Carrying out the plan is the transformation of the plan into a concrete reality and the production of a solution to the problem. Finally, looking back is the confirmation of the result and the assessment of correctness of the solution.

Most problem solving methods are generic and can be applied to a variety of domains, though some were developed to provide mathematics or science students with an explicit method for solving problems. Deek[11] developed a domain-specific model to be used in computer programming. The adaptation of this model to fit the needs of students involved in engineering design is discussed in this paper. We believe that the problem solving and program development model, used in freshman level computing courses at NJIT[12], is complementary to the engineering design pedagogy. Computer science instructors teach their students to develop the algorithms that are necessary to create useful and usable solutions and programs. These skills are also what engineering instructors want their students to know. To develop useful algorithms, students should realize that problem solving is a recursive process, where information obtained and skills gained at any one point in the process may, and often must, be revisited at other points along the way to a solution. Next, we discuss the tasks involved in each of the five stages of the problem solving process for engineering design: formulating the problem, planning the solution, designing the solution, testing the solution, and delivering the solution.

Problem formulation requires the construction of a well-defined description through refinement of the given problem statement, including diagrams, mathematical formulation, etc. The preliminary problem understanding is obtained by refining the problem description using inquiry questions. A structured representation is obtained by extracting and organizing the relevant information (goal, givens, unknowns, conditions, constraints) from the problem description. *Solution planning* requires a strategy to be discovered by first identifying alternative solutions using heuristics such as solving simpler, related or analogous problems. Problem decomposition follows where the original problem is decomposed into a collection of intermediate subproblems based on the strategy selected; these are then decomposed into sub-subproblems, and so on. Relevant data organization, based on problem decomposition, is accomplished by associating appropriate givens and unknowns with each subproblem. The next stage, *solution design*, requires the high-level plan produced by the proceeding planning stage be refined. This involves the sequencing of subproblems, the determination of whether the subproblems require further decomposition, and the establishment of hierarchical relationships among the various solution components. The sub-components are now viewed as units whose functions must be specified, and the data associated with these units (input, output, and intermediate data) are more formally represented. A final detailed design task transforms each subproblem into a corresponding solution specification. The *solution testing* stage requires generating test cases on the basis of problem requirements and certifying that the proposed solution satisfies the test results. Test results must be verified not only for correctness and completeness but also for performance criteria such as reliability, usability, etc. Modification to the design, strategy or even problem formulation may be required on the basis of testing. In addition to product testing, process evaluation and feedback is an integral part of the problem solving methods. The last stage, *solution delivery*, requires that information produced during the course of previous stages be organized and presented. The documentation of the solution strategy, design, and test results is important for subsequent refinements and updates.

We have developed and implemented a base-line study in an engineering design course to evaluate this methodology and its impact on students' problem solving abilities, skills, knowledge, and attitudes in a first-year course on engineering design.

III. Implementation

Many aspects of engineering design courses naturally benefit from the inclusion of problem solving activities. However, for students to learn how to tackle difficult problems on their own, the instructor must encourage the development of problem solving skills, and one way to accomplish this task is to use a problem solving heuristic, such as that described in the previous section. Though problem solving skills should be addressed in all components of the course, in the ED&G 100 section previously mentioned, assignments specifically designed to enhance students' problem solving skills were introduced in the design and solid modeling components. Activities related to the design project tried to emphasize the importance of the preliminary problem description and to teach students how to create a structured problem representation for the problem formulation step. The solution planning and solution design steps were implemented through brainstorming in the design component and through encouraging the students to write out solution steps before implementing them. The solution testing aspect requires students to perform calculations on their individual designs and to implement their solution on the computer for the solid modeling component. The solution delivery step involves producing professional looking reports, both individually and in groups, and giving oral presentations. Allocating enough time to give students adequate practice in all these steps in a one semester course is generally difficult, especially since problem solving skills are often given a back seat to other course objectives.

A. Solid Modeling Component

The solid modeling component of the course is ideal for giving students practice dividing a large problem into smaller segments for which a solution method can be more easily determined and tested. Solid modeling involves taking a complex solid object and modeling it on the computer using a software package, such as AutoCAD. This method requires using simple two-dimensional objects, converting them to three-dimensional solids, and manipulating them to produce the desired solution. A solid object, such as the one in Figure 1, can be split into its components (Figure 2), as is consistent with problem decomposition aspect of the solution planning step. In Figure 2, only the first three rows are complete. With proper application of this method, the bottom row of the tree would be directly translated into solid modeling commands. Though this sort of decomposition was introduced, the students preferred to express the solution in a linear fashion, as shown in Figure 3.

Given a problem, especially one to be solved on a computer, students seem to want to delve immediately into drawing objects on the computer without proper analysis or even jotting down relevant notes, such as dimensions. By having them write out the solution to a solid modeling problem before sitting down at the computer, they were forced to consider a solution strategy for the problem before attempting to create the drawing on the computer.

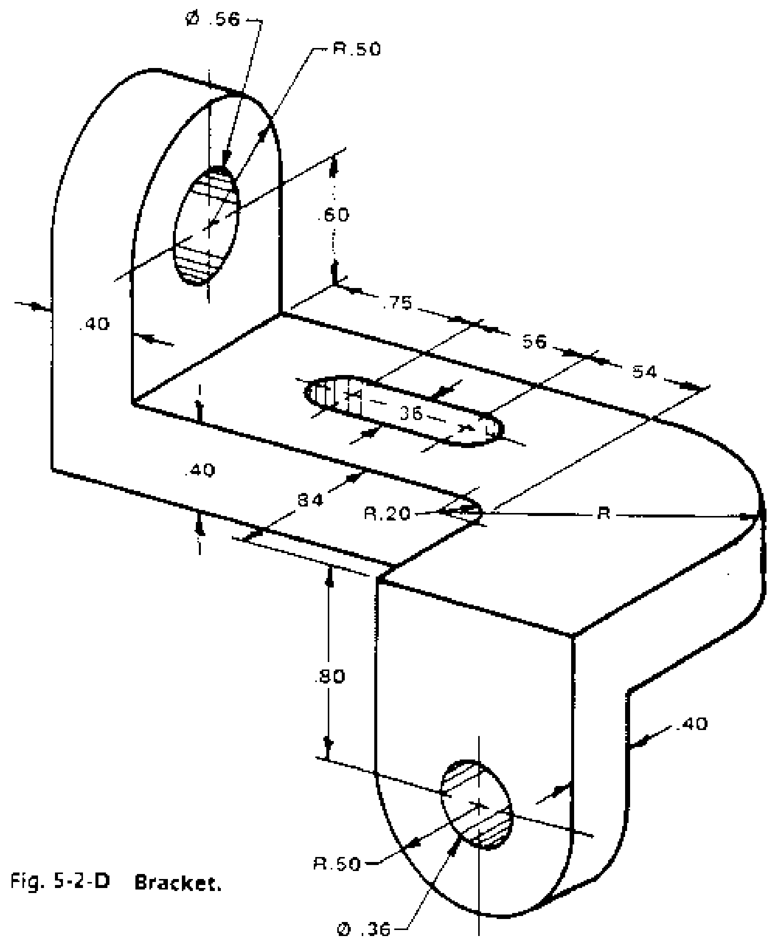

Figure 1: An example of an object[13] used in a solid modeling exercise.

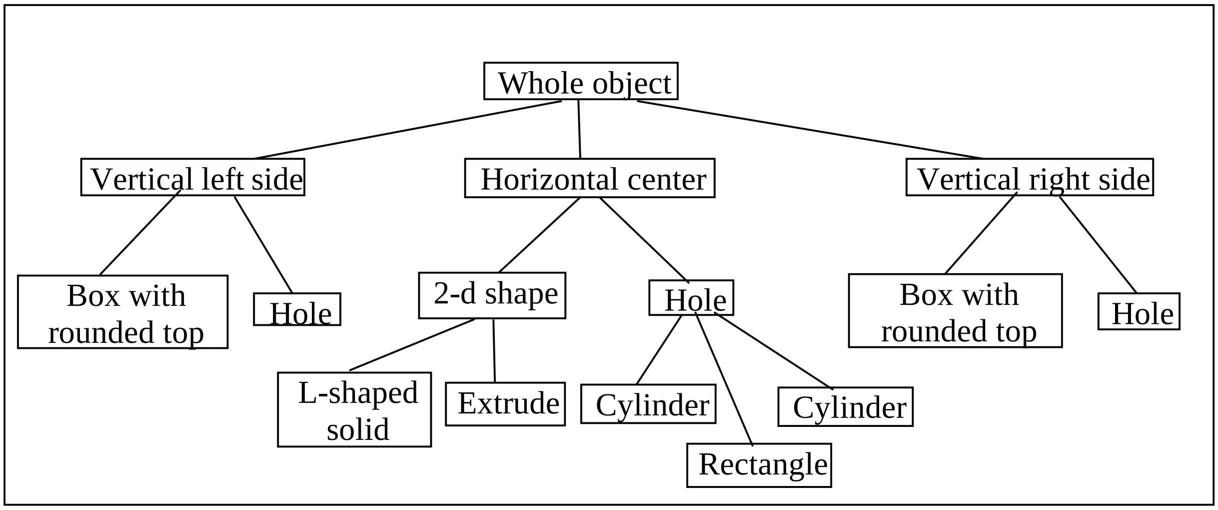

Figure 2: Solid object shown in Figure 1 partially divided into subproblems.

Example Solid Modeling Question and Answer

Outline how you would create a solid model of the bracket in Fig. 5-2-D. You do not need to include the commands you would use, but you should understand how to carry out each step in your process. All extrusions, revolutions, Boolean operations and changes in user coordinate system (UCS) should be indicated.

Answer:

1. In the world coordinate system, draw a two-dimensional profile of the horizontal portion of the bracket. Also draw a 2-d profile of the hole.
2. Extrude both the hole and the profile.
3. Subtract the hole from the profile.
4. Define a new UCS parallel to the vertical section on the left part of the bracket.
5. Draw a 2-d profile of vertical section of the bracket.
6. Extrude the 2-d profile.
7. Draw a cylinder which will become the hole.
8. Subtract the cylinder from the extruded vertical section.
9. Define a new UCS parallel to the vertical section on the right part of the bracket.
10. Draw a 2-d profile of this section of the bracket.
11. Extrude the 2-d profile.
12. Draw a cylinder which will become the hole.
13. Subtract the cylinder from the extruded section
14. Union the three sections of the bracket

Figure 3: Sample of written solution to the solid modeling problem in Figure 1.

As part of the solution testing assignment, the students were asked to discuss whether or not their written solutions were helpful and to explain how they would need to modify their preliminary solutions. Only 11 students wrote down answers to these questions. Eight of them said that writing out the solution on paper beforehand helped them develop the solution on the computer later. Two students created written and computer solutions simultaneously and then commented that the written solution was a waste of time. One student pointed out the utility of preliminary solution design for more complex problems, and another student commented that this exercise helped break down the problem into smaller components. In reality, due to the wide range of skills and experience present in the class, some of the students probably found the problem too simple to warrant spending much time on these preliminary steps. For their first attempt at this type of exercise, many of the students found that the solution they developed did not correctly correspond to the steps they needed to produce the object in the solid modeling software.

The structure of the course did not allow enough time to assign gradually more complex solid modeling problems, which would be helpful for further development of students' skills. However, as shown by the results on a quiz problem, even a few of these types of exercises can be useful. On a written quiz, students were asked to write out the solution steps to a solid modeling problem. All 17 students in the class gave a series of steps which could be directly translated into solid modeling commands. Four different solutions were proposed for the problem, 10 students gave the simplest solution (with the fewest steps), and half included some kind of labeling on the figure in the original problem to supplement their solution. Another section of the course, which was taught by a different faculty member using more traditional methods, was also presented with this same problem as an in-class assignment. Of the 15 students in that section who turned in this assignment, one student came up with the simplest solution, three students wrote solutions that were incomprehensible (as opposed to zero students in the other class), and only one-fifth labeled the figure in the original problem to illustrate their solution. In addition, many of these students' solutions did not translate directly into a series of steps to be implemented on the computer. For instance, all the students in the first section included as part of their solution that a cylinder should first be drawn and then subtracted from the rest of the object, which is how this step would be implemented in AutoCAD. In the second section, 6 students said to subtract a cylinder from the rest of the object, which would have to be translated into the two steps mentioned previously before being implemented on the computer. Finally, three students in the first section and one student in the second section wrote their answers in an outline form which clearly identifies and solves two or more subproblems.

B. Engineering Design Component

In the engineering design component of the course, the students have the opportunity to apply all the steps of the problem solving process to a problem that is not well defined. Initially, the students were asked to define the problem given unstructured assignments. The results of these assignments were largely unsuccessful. Many students did not give responses which demonstrated a proper understanding of the problem, either because they did not read the problem statement carefully enough or they did not put in the effort to consider the ideas presented in the problem statement. Though students did, to some extent, apply the entire problem solving heuristic to the design project, they required significant guidance through each of the steps to successfully understand and implement each one. Since this study describes a first attempt to implement this problem solving heuristic in an engineering design class, the amount of guidance required by students was not anticipated in advance.

First, students were asked to interpret a problem statement to identify project goals, givens and unknowns. Though students had previous experience with interpreting a problem statement through several in-class assignments, they did not have enough of the correct type of experience to properly complete this type of unstructured assignment. Many students gave answers that were often incomplete, especially for the project description. For instance, most students mentioned the stability and/or safety of the original cart but did not include specifics of why the original cart was unsafe or unstable.

By using a more guided instructional approach, similar to the guided decision-making model discussed by Woods, *et al*.,[7] the students generally provided more detailed answers. To encourage students to analyze the problem in detail and begin solution planning, the students were asked to answer the inquiry questions given in Figure 4. For instance, question A was meant to help students understand the problem description. Their answers to this question, which were more detailed than to the unstructured exercise previously described, included that the cart tips easily and is difficult to push. Questions C and D were intended as a beginning to the solution planning aspect of this project and as a tool to help students begin to relate the key points in the problem description to possible solutions. However, these inquiry questions need to be phrased properly to elicit specific answers. For question C, all students included some details relating back to the problem description in their answers, but for question D half of the responses were simply "yes". In the future, a problem solving methodology might be more successfully implemented by beginning the semester with a guided decision-making approach and teaching students to ask useful inquiry questions. Future assignments throughout the semester can gradually transition to using an unstructured approach to successfully evaluate an engineering design problem.

Group Design Project
Write-up of individual ideas

A. List the problems (at least 4) with the current material handling system.

B. List the constraints for the design of the new material handling system.

C. List each of your ideas for the new material handling system. Explain how each of your ideas addresses each of the problems listed in part A.

D. Are each of your ideas consistent with the constraints listed in part B?

Figure 4: Some inquiry questions used to guide students through implementation of the problem solving heuristic to the design process.

IV. Discussion

The primary obstacle to implementing these types of problem solving activities is the students' perception that extra work is required on each problem. Though weekly homework assignments were an important component of the class worth 10% of the grade, most students were reluctant to spend much time on assignments which involved interpreting the design project problem and performing important calculations. Even though the necessary calculations to perform for their group design project were explicitly described, only about half of the design project groups performed these calculations on their proposed design, and none of the groups compared their calculations with those on the original cart that they had done for homework. Even if given adequate class time for the exercise, the students tend to be impatient to obtain the final answer and are resistant to doing any more work than they see as absolutely necessary, even if they do see some value in it. Ideally, the students should be able to apply this problem solving heuristic to almost any problem. However, they generally could not use the problem solving heuristic unless a very specific example is first given in class. Then, they followed the example very closely rather than modifying it to meet their own needs; i.e. most students required a very structured pedagogical approach in order to complete the assignment satisfactorily.

Other obstacles to implementing a problem solving heuristic in the classroom include lack of available activities and ambiguity of assessment of problem solving skills. One of the most daunting shortcomings of most design textbooks is the lack of guided instructional problems that can be implemented in the classroom. Instructors often do not have the time to develop these types of assignments and creating a series of assignments designed to gradually develop problem solving skills is likely to require some trial and error. In addition, since problem solving is perceived as a vague skill, assessing problem solving skills can be perceived as too subjective and imperfect. Assessment methods[12] have been developed that seem to work well in assessing the ability of a student to apply a problem solving heuristic on a quiz and have shown to provide an objective and consistent grading standard.

Based on input from a post-course questionnaire, students' reaction to the focus on problem solving is mixed. The majority of students felt that their problem solving and design skills improved, and they generally believed that they gained additional understanding of problem solving and design by doing the assignments. Yet, more of the students felt that the course did not increase their interest in problem solving and design. Most of the students indicated an understanding of the importance of problem solving and design to their other technical courses and to the profession of engineering. In addition, most of the students found the course content interesting and valuable.

In order to improve the teaching of problem solving skills in ED&G 100 in the future, a guided instructional approach with assignments gradually increasing in difficulty seems likely to be successful. Though problem solving skills will be incorporated with the graphics component of the course, due to time constraints, most of the assignments will relate to the design component. Initial exercises will consist of guided inquiry questions, such as those in Figure 4, which will lead students through the decomposition of the main problem into properly defined subproblems and the association of given data with the appropriate subproblem. In addition, the specific questions Woods, et al. suggests to help students identify and associate key points in the problem description with the problem solution should help students to interpret future problems on their own. Subsequent exercises would have students respond to more general questions and develop their own inquiry questions to help guide them through the solution process. Ideally, enough practice would be given throughout the semester such that students should be able to interpret the problem statement and apply the problem solving heuristic on their own by the end of the semester. Also, requiring students to keep their notes and exercises in a problem solving notebook would assist them when reviewing examples done in class and examining previous solutions to see if any give insight into a current problem.

V. Summary

Though problem solving is considered an important component in the engineering design process, it is usually not given very much consideration in the classroom. A problem solving heuristic, which has been implemented successfully in introductory computer science classes, was integrated into the solid modeling and design components of an introductory engineering class for one semester. This problem solving method involves the following five steps: formulating the problem, planning the solution, designing the solution, testing the solution, and delivering the solution.

In the solid modeling component, students initially wanted to immediately implement a solution using the CAD software without any preliminary planning. Although they were reluctant to spend time on preliminary planning, they found that planning and designing the solution helped them to solve the problem. This class was better able to design and describe a viable solution to a solid modeling problem over a class that did not have prior exercises utilizing this problem solving heuristic.

The design component gives students experience dealing with problems that are not well defined. This study showed that many students have difficulty developing a good problem formulation in an unstructured assignment. Guided engineering instruction was used to help students deal with understanding these types of problems. In the future, in order to teach students how to deal with problems that are not well defined, design-related problem solving assignments could be given throughout the semester. Initially, the assignments would use guided engineering instruction and inquiry questions, but by the end of the semester, students should be able to interpret and explain a problem description from an unstructured assignment.

Acknowledgements

The author would like to acknowledge the suggestions and advice of the other ED&G 100 faculty at Penn State Berks, Thomas H. Gavigan and Margaret Condon-Vance.
This project was supported in part by the National Science Foundation, Award # CTS-0075076.

Bibliography

1. Richardson, J., Corleto, C., Froyd, J., Imbrie, P.K., Parker, J., & Roedel, R. "Freshman Design Projects in the Foundation Coalition." *Proceedings of the 1998 Frontiers in Education Conference,* November 4-7, 1998, Tempe, AZ.
2. Burton, J.D. & White, D.M. (1999). "Selecting a Model for Freshman Engineering Design." *Journal of Engineering Education*, 88 (3), 327-332.
3. Earl, James H. *Engineering Design Graphics*. New York. Addison-Wesley Pub. Co. (1999).
4. Burghardt, D. M. *Introduction to Engineering Design and Problem Solving*. New York. The McGraw-Hill Companies, Inc. (1999).
5. Wales, C.E., Stager, R.A., & Long, T.R. *Guided Engineering Design*. New York. West Publishing Co. (1974).
6. Ullman, David G. *The Mechanical Design Process*. New York. Mc-Graw-Hill, Inc. (1992).
7. Woods, D.R., Felder, R.M., Rugarcia, A. & Stice, J.E. (2000). "The Future of Engineering Education. III. Developing Critical Skills." *Chemical Engineering Education*, 34 (2), 108-117.
8. Woods, D.R., Hrymak, A.N., Marshall, R.R., Wood, P.E., Crowe, C.M., Hoffman, T.W., Wright, J.D., Taylor, P.A., Woodhouse, K.A., & Bouchard, C.G.K (1997). "Developing Problem Solving Skills: The McMaster Problem Solving Program." *Journal of Engineering Education*, 86 (2), 75-91.
9. URL of problem statement: http://www.ecsel.psu.edu/~dhushy/EDG100/fa98_s1/design2.html. The actual problem statement given was modified to eliminate the requirement of a prototype and the mention of primary source information, due to constraints at the campus.
10. Polya, G. *How to Solve It*. Princeton, NJ. Princeton University Press (1945).
11. Deek, F.P. *An Integrated Environment for Problem Solving and Program Development*. Unpublished Ph.D. Dissertation, New Jersey Institute of Technology. Newark NJ (1997).
12. Deek, F.P., Hiltz, S.R., Kimmel, H., & Rotter, N. (1999). "Cognitive Assessment of Students' Problem Solving and Program Development Skills." *Journal of Engineering Education*, 88 (3), 317-326.
13. Jensen, C, & Helsel, J. D. *Fundamentals of Engineering Drawing, Second Edition*. New York, McGraw-Hill Book Company (1985).

SHARI J. KIMMEL
Shari Kimmel has been teaching Engineering Design and Graphics 100 at Penn State Berks-Lehigh Valley College. She received a BS in mechanical engineering from Rutgers University and a PhD in aerospace engineering from the University of Southern California. She worked as a post-doc at Penn State University - University Park. Currently, Dr. Kimmel is a visiting research scientist and the PI on an NSF grant at Lehigh University.

FADI P. DEEK
Fadi P. Deek is Chairman of the Information Technology Program and Associate Professor of Computer and Information Science at NJIT. His current research includes learning technologies, software engineering, programming environments, problem solving/cognition/learning theory, and computer science education. He is currently Project Director for the $2.5 million N. J. Information-Technology Opportunities for the Workforce, Education and Research Project.

HOWARD KIMMEL
Howard Kimmel, Professor of Chemistry and Assistant Vice President for Academic Affairs at New Jersey Institute of Technology, has spent the past twenty-five years designing and implementing professional development programs and curricula for K-12 teachers in science and technology. At the college level, he collaborates on projects exploring teaching methodologies and assessment strategies in first year college courses in the sciences, engineering, and computer science.